

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-66835

(43) 公開日 平成7年(1995)3月10日

(51) Int.Cl.<sup>6</sup>

識別記号

庁内整理番号

F I

技術表示箇所

H 0 4 L 12/56

9077-5K

H 0 4 L 11/20

1 0 2 D

審査請求 有 請求項の数 8 O L (全 16 頁)

(21) 出願番号 特願平6-155599

(22) 出願日 平成6年(1994)7月7日

(31) 優先権主張番号 93480104.4

(32) 優先日 1993年7月30日

(33) 優先権主張国 フランス (F R)

(71) 出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション

INTERNATIONAL BUSINESS MACHINES CORPORATION

アメリカ合衆国10504、ニューヨーク州アーモンク (番地なし)

(72) 発明者 ジャン=ピエール・アルフォンシ  
フランス06610 ラ・ゴード シュマン・ド・コンプ (番地なし)

(74) 代理人 弁理士 合田 潔 (外2名)

最終頁に続く

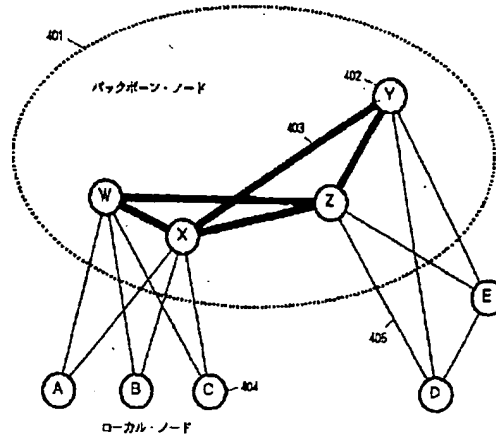
(54) 【発明の名称】 通信ネットワークおよび該ネットワークでの経路選択方法

(57) 【要約】

【目的】 現在の経路指定アルゴリズムが、最適経路を選択する前にネットワーク内のソース・ノードから宛先ノードへの可能なすべての経路を計算するため、経路計算はしばしば時間と資源を消費するという問題を解消すること。

【構成】 本発明は、ネットワークが通常は階層構造に基づいて構築されるという単純な観察を基礎とする。ユーザにとって必要な冗長性や信頼性が得られるように高度にメッシュ化された「バックボーン」を、高スループット回線で相互接続された1組のノードを使用して構築する。このバックボーン以外のノードであるローカル・ノードは、それ自体がバックボーンに結合されており、これらの属性がトポロジ・テーブルで動的に管理される。所与の接続について、限られた数のノードだけが適格と定義され、最適経路探索の際にこのアルゴリズムによって考慮される。

ネットワークの分解



1

## 【特許請求の範囲】

【請求項1】伝送リンクと相互接続された複数のノード（201ないし208）とを含むバケット交換通信ネットワークにおいて、該ネットワークのアクセス・ノードが、  
データ・バケットを受信し送信する手段と、  
ネットワーク構成要素を記憶し更新する手段とを含むとともに、

ネットワーク内に位置する各宛先ノードと通信するための1または複数の使用可能リンクを事前選択して該事前選択の情報を記憶するリンク事前選択手段と、  
各接続要求ごとに、前記記憶された事前選択リンクのうち、前記アクセス・ノードから宛先ノードに至る最適経路を決定する手段とを備える、通信ネットワーク。

【請求項2】前記記憶更新手段が、  
前記アクセス・ノードをバックボーン・ノードまたはローカル・ノードとして識別する手段と、  
前記ネットワークの他のノードについてバックボーン・ノードとローカル・ノードとを識別する手段と、  
バックボーン・リンクとローカル・リンクとを識別する手段と、  
を含む、請求項1に記載のネットワーク。

【請求項3】前記リンク事前選択手段が、  
すべてのバックボーン・リンクを、経路決定に使用可能なものとして選択する手段と、  
ノードまたは宛先ノードあるいはその両方に結合されたすべてのリンクを、経路決定に使用可能なものとして選択する手段と、  
他のすべてのリンクを、経路決定に使用可能なものとして選択しない手段とを含む、請求項1または2に記載のネットワーク。

【請求項4】前記指定経路決定手段が、修正ベルマン・フォード・アルゴリズムを計算する手段を含む、請求項1、2または3に記載のネットワーク。

【請求項5】アクセス・ノード（300）において、伝送リンク（209）と相互接続された複数のノード（201ないし208）とを含むバケット交換通信ネットワーク（200）内での指定経路を選択するための方法であって、  
ネットワーク構成要素を記憶し更新する段階を含むとともに、

ネットワーク内に位置する各宛先ノードと通信するための1または複数の使用可能リンクを事前選択して該事前選択の情報を記憶するリンク事前選択段階と、  
各接続要求ごとに、前記記憶された事前選択リンクのうち、前記アクセス・ノードから宛先ノードに至る最適バケット指定経路を決定する段階とを含む方法。

【請求項6】前記記憶更新段階が、  
前記アクセス・ノード自身をバックボーン・ノードまたはローカル・ノードとして識別する段階と、

2

前記ネットワークの他のノードについてバックボーン・ノード（402）とローカル・ノード（404）とを識別する段階と、

バックボーン・リンク（403）とローカル・リンク（405）とを識別する段階とを含む、請求項5に記載の方法。

【請求項7】前記リンク事前選択段階が、  
すべてのバックボーン・リンクを、経路決定に使用可能なものとして選択する段階と、  
ノードまたは宛先ノードあるいはその両方に接続されたすべてのリンクを、経路決定に使用可能なものとして選択する段階と、

他のすべてのリンクを、経路決定に使用可能なものとして選択しない段階とを含む、請求項5または6に記載の方法。

【請求項8】前記指定経路決定段階が、修正ベルマン・フォード・アルゴリズムを計算する段階を含む、請求項5、6または7に記載の方法。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】本発明は、大規模バケット交換ネットワークにおける高速伝送システムに関し、より詳細には、ソース・ノードと宛先ノードの間の経路選択を高速化するための装置および方法、特にネットワークをバックボーンとローカル・ノードに分解するための方法に関する。

## 【0002】

【従来の技術】データを「バケット」と呼ばれるデータの集合体として伝送するバケット交換ネットワーク手段によって、複数のデータ処理要素を相互接続することがますます有用になってきた。そのようなネットワークは、データ処理装置を支援するエンド・ノードに接続された複数の相互接続された交換ノードを含んでいる。そのようなバケット・ネットワークは、地理的に広く分布した非常に大きなものになり得る。そのような状況では、互いに通信しようとする2つのエンド・ノード間の効率的な経路選択が最も重要になる。H. ヌスバウマー(Nussbaumer)のTeleinformatique2 (pp. 92~117) Presses Polytechniques Romandes 1987にさまざまな方法が要約されている。

【0003】経路選択：ネットワーク・レベルで解決しなければならない一般的な問題は、ソース・ノードと宛先ノードの間の経路を見つけることである。ネットワークがデータグラムを使用するときは、経路選択は各バケットごとに個別に行わなければならない。仮想回線では、経路決定は回路確立時にだけ1度行われる。どちらの場合にも、経路指定アルゴリズムの選択は、しばしば相反する多くの必要条件を満たさなければならないので、容易ではない。このアルゴリズムは、ノードの実現が複雑にならないように実施できるよう簡単でなければ

ならず、ネットワーク上にどんな混乱があっても、正確なパケットの進行を保証しなければならない。トラフィックの変動やネットワーク・トポロジのいかにかわらず、このアルゴリズムは満足できる結果を提供できない。また、さまざまなエンド・ユーザが公平になるようにし、ネットワークにアクセスする権利を平等に分配しなければならない。最後に、経路指定アルゴリズムは、できれば利用タイプによって変わり得る基準に従って、ネットワークを最適に利用できるようにしなければならない。多くの場合、ネットワークは、パケット伝送時間が最短になり、かつ最大数のパケットが転送できるように実施される。その主な目的は、それぞれ伝送時間の削減とスループットの向上である。他の場合には、通信コストを削減し、あるいは破局的な回線、ノード故障、トラフィック・ピークの場合にも正確に動作（過度に性能が低下することなく）できる信頼性の高いネットワークを開発することがその目的である。さまざまな制約があるため、多数の異なるタイプの経路指定がある。そのあるものは、トラフィックの変動およびネットワーク・トポロジを統合するそれぞれの能力に応じて決定的方針（deterministic policy: 固定方針）または適応的方针（adaptive policy）に対応する。特定のノードから経路が決定される場合は、経路指定アルゴリズムを集中化することができる。他のものは全ノードに分散される。これは、信頼性の点では好ましいが、アルゴリズムがより複雑になり、経路の最適化を実施することがより難しくなる。いくつかのアルゴリズムは分類が難しいが、あふれ経路指定（flooding routing）またはランダム経路指定（random routing）と呼ばれる特別な技法を使用している。

#### 【0004】1. あふれ経路指定

あふれ経路指定は、各ノードが、入力リンクを除くそのノードのすべての出力リンク上にパケットを再送信するという非常に簡単な原理に基づくものである。K個の他のノードに接続されたノードが、受け取ったばかりのパケットのK-1個のコピーを再送信する。この方法により、ある条件でパケットの少なくとも1つのコピーを宛先ノードへ配布することが可能になり、したがってソース・ノードから宛先ノードに至る少なくとも1つの経路が存在しなければならない。この経路指定は、たとえばネットワーク内のいくつかの構成要素の破局的障害の後でネットワーク・トポロジが変化するときでも、確実に行われる。あふれ経路指定は、ネットワーク利用において非常に強い頑丈さを与える。さらに、ソース・ノードと宛先ノードの間の考えられるリンクがすべて網羅的に試され、ネットワークが過負荷でない場合、この方法は、少なくとも1つのパケットのコピーが最小の遅延でより短い経路を通して受取り側に到達するようにする。この技法は、メッセージを経路指定するための経路指定テーブルも、ネットワークにおける宛先ノードの地理的

位置の知識も必要ないので、簡単に実施することができる。受取り側は、パケットが自己宛にアドレス指定されることだけを知っていればよい。

【0005】これらの頑丈さ、簡易さおよび敏速さの品質に対する代償として、ネットワーク資源の利用度が低下し、また混雑が生じやすくなる。通常、パケット・ヘッダ内に、パケットが通過することを許されるノードの最大数を示すフィールドを置くことと好都合である。このフィールドは、パケットがノードを通過するたびに減分され、空のフィールドを有するパケットが破棄される。

#### 【0006】2. ランダム経路指定または確率的（stochastic）経路指定

この技法は、あふれ経路指定とは異なり、ノード自体のレベルで経路決定を下すためのネットワーク構造およびトラフィックについての完全な知識を有することをノードに要求しない。しかしながら、余りに大きなゴースト・トラフィックの発生を制限するために、ノードはそれ自体が受け取ったパケットのコピーをすべての出力リンク上に系統的に送ることはしない。ランダム経路指定は、任意に選択した出力リンク上に受信パケットの1つまたは複数のコピーを放出するものである。この方法は、選択的あふれとも呼ばれる。この方針は、ノード・レベルにおける非常に簡単な経路選択をもたらす。パケットの複製によるネットワークの飽和を制限する。しかし、平均経路指定遅延は決定的技法による場合よりも長くなる。パケットは、宛先ノードに向かう最短の経路ではなく遠回りの経路をとる。

#### 【0007】3. 決定的経路指定

##### （a）固定式経路指定

固定式経路指定は、ネットワーク・トポロジや通信回線上で予想される平均トラフィックなど一般のネットワーク特性にしたがって、異なるパケットがとるべき経路を定義する。経路指定規則は1回確立され、ユーザが優先する性能基準の最適化を狙いとする。ほとんどの場合、ネットワークを介する平均パケット伝送時間を最小にしなければならない。この方法は、各ノードのレベルで経路指定テーブルを作成するものである。協議により、受け取ったばかりのパケットをどの出力リンク上に転送しなければならないかを、ノードが決定できるようになる。固定式経路指定は、その原理が非常に簡単である。ノード内で行われる処理は待ち行列の管理とテーブルの読取りだけであり、最適化アルゴリズムはネットワーク設計レベルで1回だけ始動される。固定式経路選択は、ネットワークの長期的かつ総合的な最適化を狙いとするが、ランダム経路選択と比較すると、ネットワークを介するパケット伝送速度をかなり高めることができる。

##### 【0008】（b）適応的経路指定

固定式経路指定とは反対に、適応的経路指定の目的はいかなるときも最適化基準を満たすことである。たとえば回線上のトラフィックの瞬間的状態に応じて、上述のテ

ールが絶えず更新される。

#### 【0009】・集中経路指定

ネットワーク特性が絶えず変化するとき、トラフィックの変動およびトポロジに応じて経路指定テーブルを周期的に更新する責任をあるノードに割り当てることによって、経路指定を適応化することができる。この方法は、原理的には単純であり、集中経路指定と呼ばれる。その主な欠点は、大きな補助トラフィックを発生すること、およびネットワークの良好な機能をただ1つのノードに従属させてしまうことである。さまざまなノードが、それらが知覚したネットワークの状態（動作可能な隣接ノード、トラフィック…）を経路指定センターに送らなければならない。経路指定センターは各ノードにそれらの経路指定テーブルをディスペッチしなければならない。補助トラフィックは経路指定センターの周囲に集中し、このためネットワークのこの部分が飽和することになる。さらに、テーブルがすべてのノードによって同時に受け取られることはない、集中経路指定はテーブルがリフレッシュされるときにいくつかの問題を引き起こす可能性がある。1つの解決策は、テーブルの確立を各ノード・レベルに分散させることである。

#### 【0010】・ローカル経路指定

ローカル経路指定方法も分散式経路指定方法（後述）も共に、局所的に収集された情報に従って、各ノードによってそれ自体の経路指定テーブルを確立することに基づくものである。ローカル経路指定技法では、各ノードが情報を隣接ノードと交換することなくそのテーブルを作成する。最も簡単な形では、この方法は、受け取ったばかりのバケットをより短い出力待ち行列内に置き、それができるだけ迅速に伝送することからなる。ローカル・アルゴリズムは原理的に、ネットワーク内でバケットを非常に素早く循環させる傾向がある。ただし、経路はさまざまな方法で選択されるので、平均経路長は最小値とは大きく異なる。

#### 【0011】・分散式経路指定

分散式経路指定は、経路指定テーブルと遅延テーブルを更新するために、隣接ノードがトラフィックおよびネットワーク状態に関するメッセージを交換する局所的経路指定方法である。

#### 【0012】・階層式経路指定

非確率的経路指定方法では、各ノードは、そのネットワーク内のノードと同数の行と、出力リンクの数と等しい数の列とを有するテーブルを必要とする。さらに、経路指定が適応式のとき、ノードは経路指定テーブルを更新するためにメッセージを周期的に交換しなければならない。テーブル・サイズおよび制御メッセージの大きさが、ノード数とともに急激に増大し、ネットワークが10個を超えるノードを含むと許容できなくなる。この問題を解決するために、ネットワークを、図1に示すように、サブネットワークの階層に分割する。具体的には、

必要な経路指定テーブルの長さが最小になるように、最適クラスタ構造（003）を決定する。前記テーブルはサブネットワーク（003）内のノードだけを考慮しており、別のサブネットワーク内のノードにアドレス指定されたバケットは、必然的にこの遠隔サブネットワークのアクセス・ノード（005）を通過する。バケットはいくつかの必須ノード（005）を通過し、この単純化の代償は、経路指定テーブルが小さくなることにより、経路指定情報が正確でなくなり、その結果メッセージ・トラフィックの経路長が長くなることである。階層的経路指定手続きは、大規模バケット交換ネットワーク（001）で特に効果的である。経路指定テーブルが小さくなると、ノード（002）内で必要な記憶域と処理が少なくなり、かつ回線（004）における通信オーバーヘッドが少なくなる。

【0013】階層レベルの最適化の問題は、Computer Networks, Vol.1, pp.155~174 (1977年1月)に所載の、L. クラインロック(Kleinrock)およびF. カムーン(Kamoun)の論文"Hierarchical Routing for Large Networks"で検討されている。

【0014】経路指定テーブル長を削減するための主な概念は、任意のノードにおいて、近接したノードに関する完全な経路指定情報と、遠隔に位置するノードに関するより少ない情報を維持することである。これはテーブルに、近接ノードの宛先ごとに1つのエントリを提供し、遠隔ノードの1組の宛先ごとに1つのエントリを提供することによって実現される。

【0015】経路指定テーブル長の削減は、ネットワークの階層区分によって達成される。基本的に、1組のノード（002）のmレベルの階層的クラスタ化（図1）は、ノードを第1レベルのクラスタ（003）にグループ化し、次にこれを第2レベルのクラスタにグループ化し、以下同様にグループ化を続けることからなる。この動作は下から上へと続けられる。第mレベルのクラスタは、最高レベルのクラスタ（第3レベルのクラスタ）であり、したがってネットワーク（001）のすべてのノードを含む。

【0016】あいにく、テーブル長のこれらの利得はネットワーク内のメッセージ経路長の増加を伴う。その結果、より長い経路長によって引き起こされる過度の内部トラフィックのために、ネットワーク性能（遅延、スループット）が低下する。しかし、極めて大きいネットワークでは、ネットワーク経路長の著しい増加なしに、大きなテーブルの縮小が達成される。

【0017】高性能ネットワーク：データ伝送は、アプリケーションに特に焦点を合わせ、また顧客のトラフィック・プロファイルに基本シフトを組み込むことにより進展してきている。ワークステーション、ローカル・エリア・ネットワーク（LAN）相互接続、ワークステーションとスーパー・コンピュータの間の分散処理、新し

いアプリケーション、および階層ネットワークと対等ネットワーク、広域ネットワーク(WAN)とローカル・エリア・ネットワーク、音声とデータなどしばしば相反する異なる構造の統合の成長により、データ・プロファイルは、帯域幅が大きくなり、パスト的、非確率的になり、またより高い接続性を必要とする。上記のことから、チャンネル接続されたホスト、事務およびエンジニアリング・ワークステーション、端末装置並びに小さいものでは中間ファイル・サーバの間でLAN通信、音声、ビデオおよびトラフィックを搬送できる高速ネットワークの分散計算アプリケーションを支援することが強く必要とされていることは明らかである。高速マルチプロトコル・ネットワークのこの展望が、データ、音声、およびビデオ情報がデジタル・コード化され、小さなパケットに分割され、共通の1組のノードとリンクを介して伝送される、高速パケット交換ネットワーク・アーキテクチャの出現の原動力となっている。

#### 【0018】1. スループット

これらの新しいアーキテクチャの重要な要件は、実時間配布の制約を満たし、音声およびビデオの移送に必要なノードの高スループットを達成するために、端末間の遅延時間を減少させることである。リンク速度の増大に比例して、通信ノードの処理速度が増大しないと、高速ネットワークの基本課題は、各ノード内のパケット処理時間を最小にすることである。たとえば、2つのエンド・ユーザ間で音声パケットを伝送するために、一般的な100ミリ秒の遅延を満たすには次のことが必要である。

【0019】・エンド点におけるパケット化およびプレイ・アウト機能に、合計36ミリ秒必要である。

【0020】・たとえばアメリカ合衆国を横断するのに必要な固定伝播遅延時間は、およそ20ミリ秒である。

【0021】・ネットワーク中をパケットが移動する際の全ノード内処理時間として44ミリ秒残る。5ノードのネットワークでは、待ち行列時間を含めた全処理時間として各ノードは約8ミリ秒有する。10ノードのネットワークでは、各ノードは約4ミリ秒有する。

【0022】図2では、同じ制約に対する別の見方が示されている。1MIPS(毎秒100万回の命令)の有効処理速度を有するノードを取り上げてみると、処理されるパケットごとにネットワークが833000個の命令を実行しなければならない場合でも、9.6kbpsの回線は1000バイトのパケットで満たすことができる。64kbpsの回線では、ノードはパケットごとに125000回の命令を収容できる。しかしながら、OC24リンクを満たすためには、この1MIPSのノードは、パケットごとに7個の命令しか実行できない。後者の場合は、10~30MIPSの有効処理速度でも、パケットごとに70~200回の命令しか収容できない。

【0023】処理時間を最小にし、高速/低エラー率の

技法を完全に利用するために、新しい高帯域ネットワーク・アーキテクチャによって提供される転送および制御機能の大部分は、端末間方式で実行される。フロー制御および特に経路選択は、中間ノードの知識と機能の両方を減少させるネットワークのアクセス点によって管理される。

#### 【0024】2. 経路指定モード

ネットワーク内の経路指定は、次の2つの側面をもつ。

1. 所与の接続のための経路を決定する。

2. 交換ノード内でパケットを実際に交換する。

【0025】ネットワークを介する経路の決定方法は数多くある。極めて高いスループットを得るために重要なことは、着信パケットをどこに経路指定するかを、交換エレメントが極めて短時間内に判断できなければならないことである。"High Speed Networking Technology, An Introductory Survey" pp. 88~96, GG24-3816-00 ITSC Raleigh March 1992に述べられているように、最もよく使用される経路指定モードは、ソース経路指定(ソース・ルーティング)およびラベル・スワッピングである。

#### 【0026】・ソース経路指定

ソース経路指定は、無接続ネットワーク用の分散経路指定の特定の実施態様である。ソース・ノード(またはアクセス・ノード)が、パケットがネットワークを介して取るべき経路を計算する責任を負う。経路指定フィールドが送られたすべてのパケットに付加され、中間ノードがそのフィールドを使ってパケットを宛先に導く。この技法では、送信ノードがネットワーク・トポロジを知らなければならない、最適経路を見つけることができない。しかし経路が決定された後は、中間ノードは経路指定を行うためにどのテーブルやパラメータも参照する必要がない。経路指定の次の段階はまさにパケット・ヘッダで行われる。この方法の欠点は、パケット・ヘッダ内の経路指定フィールドが、ある記憶域を占有し、オーバーヘッドとなることである。しかしこれはまったく些細なことであり、高速に経路指定を決定できるという利点が、帯域オーバーヘッドのわずかな増加という欠点よりも勝っている。すべての経路指定情報が各パケット内に含まれるので、端末間通信を提供するために経路に沿ったセットアップを行う必要はない。したがって、データグラム・サービスから利益を受けるデータ・アプリケーションは、この技法によって有効に支援される。ただしデータグラム・トラフィックは、各データグラムを1つの接続と見ることができ、経路の計算を必要とするということも特徴とする。各パケットは、自己完結型ユニットとしてネットワークを介して経路指定され、他のすべてのパケットと無関係である。ソース・ノードに関する重要な点は、各データグラムについて、より短時間で最適の経路を決定することである。

#### 【0027】・ラベル・スワッピング

ラベル・スワッピングは、接続指向ネットワーク用の分

散経路指定の特定の実施態様である。このネットワークは、一般的に、何らかの形の「チャンネル」を使用するリンク上で、多数の接続（セッション）を多重化する。リンク上を送信される各パケットは、このパケットが属する論理接続を識別する任意の番号を含むヘッダを有する。たとえば、各パケットは、そのパケットが送られる伝送リンクを識別するために中間ノードによって使用されるラベルを、その経路指定フィールド内に含むことができる。パケットがノードに到達すると、そのヘッダからラベルが抜き出され、リンク・アドレスに加えて新しいラベルを提供する経路指定テーブルを索引するために使用される。新しいラベルは古いラベルに上書きされ、パケットはそのリンク・アドレスを使用して転送される。経路指定テーブル内の情報は、接続セットアップ時に更新される。経路上の各ノードは、各接続方向用のラベルを選択し、それを隣接ノードに送る。

【0028】呼出しセットアップおよび資源予約処理は、次のような段階を含む。

【0029】・接続要求は、ユーザにより起点ネットワーク・アドレスと宛先ネットワーク・アドレスならびにデータ・フロー特性を含むパラメータ・セットを使って指定される。

【0030】・経路決定は、ソース・ノードによってその局所的経路指定トポロジ・データベースから実現される。

【0031】・経路予約は、指定されたノードに沿って、特別なメッセージに入れて送出される。このメッセージは、通常は前述のソース経路指定技法に従って経路指定され、トラフィック・タイプが必要とするサービス・レベルを提供するため、その接続テーブルをセットアップし、その資源を予約する（たとえば、経路の各リンク上の帯域幅予約）よう、ノードに指示する。

【0032】・テーブル更新は、ネットワーク資源の使用可能性の変化を反映する。その情報は、ネットワーク内のすべてのノードに送られる。

【0033】ラベル・スワッピング技法では、接続テーブルを動的にセットアップし維持する必要がある。これは、新しい接続が確立されまたは古い接続が終了されるとき、テーブルが更新されることを意味する（もちろん、ネットワーク・トポロジのデータベースは完全に独立して維持される）。これは、接続セットアップが比較的たまにしか行われず、あまり時間が重要ではない場合に可能になる。この制限のため、データグラム移送はまったく非効率的である。しかしながら、実時間音声接続のような非常に短いパケットしか必要としない接続は、パケットのオーバーヘッドが低いので、この技法により有効に支援される。接続がいったん確立された後は、パケットが送られるたびに宛先アドレスをヘッダ内に置く必要はない。このパケットのためにどの接続を使用するかを指定するフィールドだけが必要とされる。

### 【0034】3. 経路選択基準

パケット・ネットワーク内のノード間の経路を選択するための主な基準の1つは、最小ホップ・カウントと最小経路長である。ホップ・カウントは、2つのエンド・ノード間で経路を作成するために使用されるリンクの数である。経路長は、2つのエンド・ノード間の経路によってもたらされる全体的伝送遅延時間の関数である。大部分の高速ネットワークでは、このネットワークを介する最悪ケースの遅延がほぼ常に許容されるので、遅延（経路長）は大きな問題ではない。しかし、ホップ・カウントは、所与の経路を実施するために必要な資源量の直接の尺度であり、したがって経路を選択する際にかなり重要である。ネットワーク・リンクが混雑すると、必然的に大きいホップ・カウントの経路が選択されるので、選択された経路は最小ホップ・カウントの経路である必要がないことに留意されたい。しかし、過大な量のネットワーク資源が1つの経路にコミットされる結果、他の経路が一層混雑し、他の接続用に必然的にさらに長いホップ・カウントの経路が選択されることになるので、そのような長い代替経路を無制限に延ばすことは許されない。それによって、長期的なネットワークのスループットが悪影響を受ける可能性がある。したがって、問題は、最小ホップ・カウント、すなわちネットワーク資源量を過度に利用しない最小経路長を有する、起点ノードと宛先ノード間の経路を選択することである。

### 【0035】4. 重要な要件

接続指向経路指定モードと無接続経路指定モードのどちらもサポートする大規模高速パケット交換ネットワークにおける分散経路指定機構では、性能および資源消費量に関して、次のように要約されるいくつかの要件が暗黙に課される。

【0036】・ソース・ノード（またはソース・ノードのために経路計算を行うノード）は、着信パケットをどこに経路指定するかを非常に短期間に決定できなければならない（各接続要求のための最適経路を計算するための計算が十分に早くなければならない）。

【0037】・中間ノードにおける交換時間が最短でなければならない（最短処理時間）。

【0038】・選択された経路に沿ったネットワーク資源を、最小ホップ・カウントの基準に従って最適化しなければならない。

【0039】・ネットワークが過負荷にならないように、制御メッセージをできるだけ制限しなければならない。

【0040】高速ネットワークでは、制御メッセージの交換によって生成される経路指定テーブルを更新するためのコストは、これが接続セットアップ前に実行される限りクリティカルではない。回線スループットが極めて高いと、通信回線自体に著しい負担をかけずに経路指定情報の交換が許される。これらの考慮から、L. クライ

ンロックとF. カムーンが提唱した階層的経路指定方法とは反対に、最適経路の基準を犠牲にせず、ノード内で維持される経路指定テーブルのより良好なアクセスおよび使用が示唆される。

#### 【0041】

【発明が解決しようとする課題】現在の所、経路指定アルゴリズムは、最適経路を選択する前にネットワーク内のソース・ノードから宛先ノードへの可能なすべての経路を計算する。ネットワーク・トポロジに関する仮定が行われず、経路計算はしばしば時間と資源を消費する。たとえば、ネットワークの特定の地理的構成のために、計算された経路が許容されないこともある。

【0042】現実には、大規模伝送ネットワークは完全にメッシュ化 (mesh) されてはいない。本発明は、ネットワークが通常は階層構造に基づいて構築されるという単純な観察を基礎としている。ユーザにとって必要な冗長性や信頼性が得られるようにするためには高スループット回線で相互接続された1組のノードを使用して、高度にメッシュ化された「バックボーン」を構築する。他のノードまたは「ローカル」ノードは、1つまたは複数のバックボーン・ノードに接続される。構成時に各ノードについてその属性がバックボーンかローカル・ノードかを定義するのは、ネットワーク設計者の責任である。ノード属性のリストはトポロジ・テーブル内に生じ、ノードがネットワークに追加されまたはネットワークから削除されるたびに更新される。経路指定アルゴリズムは、特定のネットワーク・トポロジを利用して経路計算の複雑さを大幅に減少させることができる。所与の接続について、限られた数のノードだけが適格とされ、最適経路探索においてアルゴリズムによって考慮される。本発明の目的は、経路選択を高速化するためにネットワークをバックボーンとローカル・ノードに分離することである。

#### 【0043】

【課題を解決するための手段】本発明は、伝送リンクと相互接続された複数のノードとを含むバケット交換通信ネットワークにおいて、該ネットワークのアクセス・ノードが、データ・バケットを受信し送信する手段と、ネットワーク構成要素を記憶し更新する手段とを含むとともに、ネットワーク内に位置する各宛先ノードと通信するための1または複数の使用可能リンクを事前選択して該事前選択の情報を記憶するリンク事前選択手段と、各接続要求ごとに、前記記憶された事前選択リンクのうち、前記アクセス・ノードから宛先ノードに至る最適経路を決定する手段とを備えた、通信ネットワークのためのネットワーク・アクセス・ノードである。

#### 【0044】

【実施例】図3に示すように、通信システムの典型的なモデルは、専用回線、キャリア提供サービスまたは公衆データ・ネットワークを使用して、高性能ネットワーク

(200)を介して通信する、いくつかのユーザ・ネットワーク(212)からなる。各ユーザ・ネットワークは、企業サーバ(213)として使用される大型コンピュータ、LAN(ローカル・エリア・ネットワーク214)に接続されたワークステーションまたはパーソナル・コンピュータを使用するユーザ・グループ、アプリケーション・サーバ(215)、PBX(構内交換機216)またはビデオ・サーバ(217)を相互接続するリンク(211)ならびに通信プロセッサの組として記述される。これらのユーザ・ネットワークは、異なる施設に分散され、広域転送設備を介して相互接続する必要がある。データ転送を編成するために異なる方法が使用できる。あるアーキテクチャは、各ネットワーク・ノードでデータ保全性に関する検査を必要とし、そのため伝送が遅くなる。他のアーキテクチャは、基本的に高速データ転送を追求するもので、最終宛先に向かって流れるパケットができる限り最高速度で処理されるように、ノード内の経路指定技法と交換技法が最適化される。本発明は本質的に後者のカテゴリに属しており、より詳細には以下の節で詳しく述べる高速バケット交換ネットワーク・アーキテクチャに属する。

【0045】高速バケット交換ネットワーク：図3における全体図は、8個のノード(201-208)を備える高速バケット交換伝送システムを示し、各ノードはトランク(209)と呼ばれる高速通信回線によって相互接続されている。ユーザによる高速ネットワークへのアクセス(210)は、周辺に配置されたアクセス・ノード(202-205)を介して実現される。これらのアクセス・ノードは1つまたは複数のポートを含み、各ポートはそれぞれ、標準インターフェースを支援する外部装置をネットワークに接続し、ネットワークを横切ってユーザのデータ・フローを他の外部装置からまたは他の外部装置へ移送するために必要な変換を実行するためのアクセス点を提供する。たとえば、アクセス・ノード202は、構内交換機(PBX)、アプリケーション・サーバ、および3つのポートを通るハブにそれぞれインターフェースし、隣接する通過ノード201、208および205によってネットワークを介して通信する。

【0046】交換ノード：各ネットワーク・ノード(201-208)は、着信データ・バケットが発信トランク上で隣接通過ノードに向って選択的に経路指定される、経路指定点を含む。そのような経路指定の決定は、データ・バケットのヘッダ内に含まれる情報に従って行われる。基本的なバケット経路指定機能に加え、ネットワーク・ノードは次のような補助的サービスをも提供する。

- ・ノード内から発するバケット用の指定経路の決定。
- ・ネットワーク・ユーザおよび資源に関する情報の検索や更新などのディレクトリ・サービス。
- ・リンク使用状況情報を含む物理的ネットワーク・トポ

13

ログの一貫したビューの維持。

・ネットワークのアクセス点での資源の予約。

【0047】各ポートは複数のユーザ処理装置に接続され、各ユーザ装置は、別のユーザ・システムに伝送されるデジタル・データのソース、または別のユーザ・システムから受け取ったデジタル・データを消費するデータ受信側、あるいは通常はその両方を含む。ユーザ・プロトコルの解釈、パケット・ネットワーク(200)上で伝送するために適切にフォーマットされたパケットへのユーザ・データの変換、およびこれらのパケットを10 経路選択するためのヘッダの生成が、ポートで走行するアクセス・エージェントによって実行される。このヘッダは、制御フィールドと経路指定フィールドからなる。

【0048】・経路指定フィールドは、アドレス指定された宛先ノードにネットワーク(200)を介してパケットを経路指定するために必要な情報をすべて含む。

【0049】・制御フィールドは、とりわけ、経路指定フィールドを解釈する際に使われるプロトコルのコード化された識別を含む(ソース経路指定、ラベル・スワッピングなど)。

【0050】経路指定点：図4は、図3に示したネットワーク・ノード(201-208)中に見られるような、典型的な経路指定点(300)の全体的ブロック図を示す。経路指定点は、高速パケット・スイッチ(302)を含み、経路指定点に到達したパケットがそこに入る。そのようなパケットは以下のようにして受信される。

・高速伝送リンク(303)を経てトランク・アダプタ(304)を介して、他のノードから受信する。

・ポート(301)と呼ばれるアプリケーション・アダプタを介して、ユーザから受信する。

【0051】アダプタ(304、301)は、パケット・ヘッダ内の情報を使用して、どのパケットをスイッチ(交換機：302)によってローカル・ユーザ・ネットワーク(307)に向けてまたはノードから出る伝送リンク(303)に向けて経路指定するかを決定する。アダプタ(301と304)は、スイッチ(302)に入る前または後にパケットを待ち行列に入れるための待ち行列回路を含む。

【0052】経路推定制御装置(305)は、ユーザによって指定された所与の1組のサービス品質を満たし、通信経路を完成させるために用いられるネットワーク資源の量を最小にするために、ネットワーク(200)を介する最適経路を計算する。次に、経路指定点で生成されたパケットのヘッダを構築する。最適化基準には、中間ノードの数、接続要求の特徴、経路内のトランクの容量と使用度が含まれる。経路指定のために必要な、ノードおよびノードに接続された伝送リンクに関する情報はすべて、ネットワーク・トポロジ・データベース(306)内に含まれる。定常状態のもとでは、すべての経路 50

14

指定点は同じネットワークのビューを有する。ネットワーク・トポロジ情報は、ネットワークに新しいノードが追加されて新しいリンクが活動化されたとき、リンクまたはノードがドロップされたとき、あるいはリンクの負荷が著しく変化したときに更新される。そのような情報は、その資源が接続されたネットワーク・ノードから発し、制御メッセージによって他のすべての経路制御装置との間で交換され、経路計算に必要な最新のトポロジ情報を提供する(そのようなデータベースの更新は、ネットワークのエンド・ユーザ間で交換されるデータ・パケットと非常に類似したパケット上で搬送される)。継続的な更新を通じてネットワーク・トポロジがあらゆるノードで最新の状態に維持されるので、エンド・ユーザの論理接続(セッション)を乱さずに、動的なネットワークの再構成が可能になる。

【0053】パケット経路指定点への着信伝送リンクは、ローカル・ユーザ・ネットワークの外部装置からのリンク(210)、または隣接するネットワーク・ノードからのリンク(トランク：209)を含む。どちらの場合でも、経路指定点は同じように動作して、パケット・ヘッダ内の情報によって指示されるように、各データ・パケットを受け取り、別の経路指定点に転送する。高速パケット交換ネットワークは、単一パケットの伝送継続中以外はどの伝送機構やノード機構をも通信経路専用10 にせずに、任意の2つのエンド・ユーザ・アプリケーション間の通信を可能にするように働く。このようにして、各通信経路ごとに伝送リンクを専用にする場合よりも、ずっと多くのトラフィックが搬送できるように、パケット・ネットワークの通信機構の利用度が最適化される。

#### 【0054】ネットワークの分解

##### 1. 修正ベルマン・フォード・アルゴリズム

ネットワークにおける最適経路を計算することが可能なアルゴリズムはいくつかある。修正ベルマン・フォード・アルゴリズム(Modified Bellmann-Ford Algorithm)は、高速パケット交換ネットワークで最も一般に使用されているものの1つである。欧州特許出願第93480030.1号、"Methods for Apparatus for Optimum Path Selection in Packet Transmission Networks"に記載されているように、このアルゴリズムは、各接続要求について下記のパラメータを用いて呼び出される。

- ・ソース・ノード・アドレス
- ・宛先ノード・アドレス
- ・その接続に必要なサービスの品質。たとえば、
- ・最大呼出しセットアップ遅延時間(実時間で処理される接続の場合に非常に重要なパラメータ)。
- ・最大端末間通過遅延時間。
- ・最大情報損失、エラー確率等。

【0055】このアルゴリズムは、サービス要件の品質を満たすような、宛先ノードとの最小ホップおよび経路



長を決定するために、ソース・ノード（またはアクセス・ノード）において、ネットワーク・トポロジのローカル・コピーを使用する。前述のように、修正ベルマン・フォード・アルゴリズムでは、ネットワークの地理的構成に関する仮定を行わず、ネットワークが完全にメッシュ化または階層化されても、同じ複雑さを必要とする。本発明の目的は、所与の接続に関して、経路計算に必要な適格ノードの数を削減することにより、ネットワーク・トポロジを簡略化することである。

\*

$$C = N \times 1 \cdot 1 (N-1) / N)^0 + (\text{第1ホップ}) \\ 2 \cdot 1 (N-1) / N)^1 + (\text{第2ホップ}) \\ 3 \cdot 1 (N-1) / N)^2 + (\text{第3ホップ}) \\ \dots \dots \dots (i+1) \cdot 1 (N-1) / N)^i + (\text{第} i+1 \text{ホップ})$$

$$C = \text{hopt} \cdot N \cdot l$$

$$= \text{hopt} \cdot N \cdot 2 \cdot L / N = \text{hopt} \cdot 2L \quad \text{ただし:}$$

hopt = 最適経路のための平均ホップ・カウント

l = 1 ノードあたりの平均リンク数

N = ネットワーク内の総ノード数

L = ネットワーク内の総リンク数

【0057】複雑さはネットワーク内のリンクの総数に直接関係する。2ノード間の所与の経路探索について、経路の計算に使用できるリンク数を事前に選択することにより、複雑さを極めて大きな割合で減少させることができる。この事前選択は、最小ホップ数の基準を同じ割合で減少させない場合、すなわちhoptが一定のままの場合に効率的である。

#### 【0058】3. ノードおよびリンクの属性

本発明の基本的な概念は、所与のノードにおいて別のノードへの経路を構築するために考慮しなければならない物理リンクを事前選択し、経路選択時に前記事前選択されたリンクを使用することである。本発明はまた、リンク・テーブルを初期化し維持する方法についても詳しく述べる。

【0059】いくつかのノードがバックボーンを構築するために選択される。その他のノードはローカル・ノードと呼ばれ、このバックボーンの少なくとも1つのノードに接続される。ローカル・ノードとバックボーン・ノードは両方とも、何の制限もなしにポート及びトランクを支援することができる。ノードの属性はトポロジ・データベース内に記録され、ネットワークにおける変化、ノードの追加またはノードの抑制ごとに更新される。ノード属性に基づき、以下の規則に従って、各リンクは新しい属性によって修飾される。

#### 【0060】

#### \* 【0056】2. アルゴリズムの複雑さ

所与のノードから他のすべてのノードへ到達するために必要な合計ホップ・カウントが、最小ホップ基準による経路選択の複雑さの測度Cを与える。Cは、ノードあたりの平均リンク数l、ネットワーク内のノードの合計数N、および最適ホップの平均数“hopt”に正比例する。

【数1】

ノード i	ノード j	リンク i j 型
バックボーン	バックボーン	バックボーン
バックボーン	ローカル	ローカル
ローカル	バックボーン	ローカル
ローカル	ローカル	ローカル

20 【0061】これらのリンク属性は、ネットワーク設計者によって定義されない。それらはノードiおよびノードjのタイプに応じて、各ノードによって動的に構築される。ノードの分類は、次の方法のうちの1つに従って行うことができる。

【0062】・小さなネットワークの場合は手動による。ソース・ノードから宛先ノードに至る経路の探索はバックボーン・レベルだけに制限され、ローカル・ネットワークを介する可能な探索はすべて除外される。

【0063】・可能な各ソースおよび宛先ごとに経路選択アルゴリズムをオフラインで使用する。

【0064】4. トポロジ・データベース: トポロジ・データベースはあらゆるノードで複製され、ネットワーク・ノードおよびリンクに関する情報を含んでいる。トポロジ・アルゴリズムは、すべてのノードにおいて、ネットワークの一貫したビューを維持するために使用される。トポロジ・データベースは次の2種類の情報を含む。

・ノードおよびリンクの物理的特性などの静的情報を含むネットワークの物理的トポロジ。

・リンクの使用度。

【0065】最初のクラスのパラメータだけが本発明に関係する。

#### 【0066】(1) 仮定

・ネットワークがN個のノードとL個のリンクを含む。  
・すべてのノードがネットワーク・トポロジに関する同じ知識を有する。

#### 【0067】(2) 定義

所与のノード(ノードi)において、任意の他のノード(ノードj)に到達するためにどのリンクを使用すべきかを決定できる単純な構造を構築しなければならない。

この行列構造は、次のように定義することができる。

【0068】・ネットワークの各リンクを表す行と各ノードを表す列がある。そのように定義された構造は、「経路選択行列」と呼ばれる $L \times N$ 行列である。

【0069】・行列要素 $E_{ln}$ は次のようなブール値である。

・ $E_{ln}=1$  (真) は、ノード $l$ からノード $n$ に到達するためにリンク $l$ が使用できることを意味する。

・ $E_{ln}=0$  (偽) は、ノード $l$ からノード $n$ に到達するためにリンク $l$ が使用できないことを意味する。

【0070】(3) 行列の初期化

行列は、各ノードにおいて独立に、次の原理に従って構成時に定義される。

【0071】(a) リンク (または行列の行)

・ノード $l$ からバックボーン・ノードに至るリンクはすべて、(ノード $l$ 自体を除く) 他のノードに到達するために使用可能である。

・任意のバックボーン・ノード間のリンクはすべて、ネットワーク内の任意の宛先に到達するために使用可能である。

・他のローカル・リンク (ノード $l$ に直接接続されない) は、それが接続されているローカル・ノードに到達するためにだけ使用可能である。

【0072】(b) ノード (または行列の列)

・ノード $j$ 列において、バックボーン・リンクはすべて使用可能である。

・ノード $j$ 列において、ノード $l$ またはノード $j$ あるいはその両方に接続されたリンクはすべて使用可能である。

・ノード $j$ 列において、他のすべてのリンクは使用可能でない。

【0073】(4) 行列の更新

新しいノードまたはリンクが追加されるたびに、ネットワークのあらゆるノードにあるトポロジ・データベースを更新しなければならない。これは、新しいトポロジ、特に新しいノードの属性を含む制御メッセージによって行われる。行列は、次の規則に従って更新される。

【0074】(a) 新しいローカル・ノード $k$ を追加するには、次のような新しい列を追加する必要がある。

・バックボーン・リンクと、ノード $l$ またはノード $k$ あるいはその両方に接続されたリンクでは値1 (真)。

・他のすべてのリンクでは値0 (偽)。

【0075】(b) 新しいバックボーン・ノード $k$ を追加するには、次のような列を追加する必要がある。

・バックボーン・リンクと、ノード $l$ に接続されたリンクでは値1 (真)。

・他のすべてのローカル・リンクでは値0 (偽)。

【0076】(c) 新しいローカル・リンク $ljk$ を追加するには、次のような新しい行を追加する必要がある。

・リンクがノード $l$ に直接接続されない場合、ローカル・ノード $j$ または $k$ では値1 (真)。

・リンクがノード $l$ に直接結合される場合、すべてのノード (ノード $l$ を除く) で値1 (真)。

・他のすべてのノードでは値0 (偽)。

【0077】(d) 新しいバックボーン・リンク $ljk$ を追加するには、行列中に、(ノード $l$ を除く) すべてのノードで値1 (真) の新しい行を追加する必要がある。

10 【0078】5. 最適経路探索の方法

本発明による方法は、以下のように要約することができる (図9)。

【0079】・段階1: ノードが属性により特徴づけられる。すなわち、ノードがバックボーン・ノードまたはローカル・ノードとして定義される。

【0080】・段階2: ノードが、ネットワーク構成をそのトポロジ・データベース内に記憶する。このデータベースは、ネットワーク内のすべてのノード間で交換される制御メッセージによって初期化され維持される。

20 【0081】・段階3: トポロジ・データベース内に記憶された情報から、ノードがバックボーン・ノードとローカル・ノードを識別する。

【0082】・段階4: ノードが、ノード属性に応じて各リンクの属性を決定する。

【0083】・段階5:  $L \times N$ 行列を構築することにより、ノードがネットワーク内の各宛先ノード用の使用可能リンクを選択する。

30 【0084】・段階6: 各接続要求ごとに、1組の事前選択された使用可能リンクについて経路指定アルゴリズムが計算される。

【0085】・段階7: 接続時間中、事前に計算した経路に沿って、データ・パケットが宛先ノードに送られる。

【0086】例: 図5に示すように、ネットワークは、構成時に、複数のバックボーン・ノード (401) およびローカル・ノードに分解される。

【0087】

バックボーン・ノード (402): W, X, Y, Z

ローカル・ノード (404): A, B, C, D, E

および

バックボーン・リンク (403):  $L_{wx}, L_{wz}, L_{xy}, L_{xz}, L_{yz},$

ローカル・リンク (405):  $L_{aw}, L_{ax}, L_{bw}, L_{bx}, L_{cw}, L_{cx}, L_{dy}, L_{dz}, L_{de}, L_{ey}$

$L_{ez}$

【0088】ノードAに関する対応する行列を次に示す。

【表1】

	19									
ノード	A	B	C	D	E	F	G	H	I	J
リンク										
Lax	0	1	1	1	1	1	1	1	1	1
Lbx	0	1	1	1	1	1	1	1	1	1
Lbw	0	1	0	0	0	0	0	0	0	0
Lbz	0	1	0	0	0	0	0	0	0	0
Lcw	0	0	1	0	0	0	0	0	0	0
Lcx	0	0	1	0	0	0	0	0	0	0
Ldy	0	0	0	1	0	0	0	0	0	0
Ldz	0	0	0	1	0	0	0	0	0	0
Lde	0	0	0	0	0	0	0	0	0	0
Ley	0	0	0	0	1	0	0	0	0	0
Lez	0	0	0	0	1	0	0	0	0	0
Lwx	0	1	1	1	1	1	1	1	1	1
Lwz	0	1	1	1	1	1	1	1	1	1
Lxy	0	1	1	1	1	1	1	1	1	1
Lxz	0	1	1	1	1	1	1	1	1	1
Lyz	0	1	1	1	1	1	1	1	1	1

【0089】図6に、ネットワーク内に、次のようなり  
ンクを有するローカル・ノードFとバックボーン・ノード\*

Lax	0	1	1	1	1	1	1	1	1	1
Lbx	0	1	1	1	1	1	1	1	1	1
Lbw	0	1	0	0	0	0	0	0	0	0
Lbz	0	1	0	0	0	0	0	0	0	0
Lcw	0	0	1	0	0	0	0	0	0	0
Lcx	0	0	1	0	0	0	0	0	0	0
Ldy	0	0	0	1	0	0	0	0	0	0
Ldz	0	0	0	1	0	0	0	0	0	0
Lde	0	0	0	0	0	0	0	0	0	0
Ley	0	0	0	0	1	0	0	0	0	0
Lez	0	0	0	0	1	0	0	0	0	0
Lwx	0	1	1	1	1	1	1	1	1	1
Lwz	0	1	1	1	1	1	1	1	1	1
Lxy	0	1	1	1	1	1	1	1	1	1
Lxz	0	1	1	1	1	1	1	1	1	1

Lyz	0	1	1	1	1	1	1	1	1	1
-----	---	---	---	---	---	---	---	---	---	---

Lfx	0	0	0	0	0	0	0	0	0	1	0
Lfz	0	0	0	0	0	0	0	0	0	1	0
Ltx	0	1	1	1	1	1	1	1	1	1	1
Lty	0	1	1	1	1	1	1	1	1	1	1
Ltw	0	1	1	1	1	1	1	1	1	1	1

Lat	0	1	1	1	1	1	1	1	1	1	1
Lab	0	1	1	1	1	1	1	1	1	1	1

【0092】記法を簡略化するため、 $T_i(k)$  (ただし、 $k=1, \dots, L$ ) が次のように示されるLエントリ・テーブルとしてノードiの行列を表現する。

リンク	TA	TB	TC	TD	TE	TF	TT	TW	TX	TY	TZ
Lax	1	A	A	A	A	A	A	A	A	A	A

\*ドTの2つのノードを追加する例を示す。

【0090】

新しいバックボーン・リンク:  $Ltx, Lty, Ltw$

新しいローカル・リンク:  $Lfx, Lfz, Lat, Lab$

【0091】行列は下記の表のように更新される。

【表2】

ノード	A	B	C	D	E	F	G	H	I	J	F	T
リンク												

【0093】・リンクkが任意の経路に使用可能なときは、 $T_i(k)=1$ である。

・リンクkが任意の経路に使用可能でないときは、 $T_i(k)=0$ である。

・リンクkが、ノードiからノードj ( $j=1, \dots, N$ , ただしiを除く) に至る経路上でだけ使用可能であるときは、 $T_i(k)=j$ である。

【0094】この例では、ノードAないしZにおけるテーブル $T_i$ は次のような値を有する。

【表3】

21

22

Lax	1	A	A	A	A	A	A	A	A	A	A
Lbw	B	1	B	B	B	B	B	B	B	B	B
Lbx	B	1	B	B	B	B	B	B	B	B	B
Lcw	C	C	1	C	C	C	C	C	C	C	C
Lcx	C	C	1	C	C	C	C	C	C	C	C
Ldy	D	D	D	1	D	D	D	D	D	D	D
Ldz	D	D	D	1	D	D	D	D	D	D	D
Lde	0	0	0	E	D	0	0	0	0	0	0
Ley	E	E	E	E	1	E	E	E	E	E	E
Lez	E	E	E	E	1	E	E	E	E	E	E
Lwx	1	1	1	1	1	1	1	1	1	1	1
Lwz	1	1	1	1	1	1	1	1	1	1	1
Lxy	1	1	1	1	1	1	1	1	1	1	1
Lxz	1	1	1	1	1	1	1	1	1	1	1
Lyz	1	1	1	1	1	1	1	1	1	1	1
Lfx	F	F	F	F	F	1	F	F	F	F	F
Lfz	F	F	F	F	F	1	F	F	F	F	F
Ltx	1	1	1	1	1	1	1	1	1	1	1
Lty	1	1	1	1	1	1	1	1	1	1	1
Ltw	1	1	1	1	1	1	1	1	1	1	1
Lat	1	A	A	A	A	A	A	A	A	A	A
Lab	B	A	0	0	0	0	0	0	0	0	0

【0095】ネットワーク内の2ノード間の経路は、ホップ数が最小になる場合に最適と見なされる。すべての最適経路は、ソース・ノードをルートとするツリーを作成する。図7に、ノードAから確立されるすべての最適経路を示す。第1レベルにはソース・ノードに隣接するすべてのノードが配置され、次に第2レベルには第1レベルのノードに隣接するまだ配置されていないすべてのノードが配置され、以下同様にすべてが終わるまで続く。図8に、本発明によるノードAからノードEへの経路探索を示す。前述のように、ノードEに到達するための適格リンクはテーブルTAで定義される。

【0096】

適格	非適格
リンク	リンク
TA	TA
Lax 1	Lbw B
Lax 1	Lbx B
Lat 1	Lcw C
Ley E	Lcx C
Lez E	Ldy D

\*

Lwx	1	Ldz	D
Lwz	1	Lde	0
Lxy	1	Lfx	F
Lxz	1	Lfz	F
Lyz	1	Lab	B
Ltx	1		
Lty	1		
Ltw	1		

30

【0097】バックボーン・リンク、およびソース・ノードと宛先ノードをバックボーンに接続するローカル・リンク（701：実線）だけが、探索プロセスで考慮される。他のリンク（702：破線）は、経路指定アルゴリズムの計算（たとえば、ベルマン・フォード・アルゴリズム）の複雑さを削減する経路選択に関与しない。この例では、4つの経路AWZE、ATYE、AXZEおよびAXYEが、3ホップで最小ホップ制約を満たす。

【0098】図6に示したネットワークの複雑さは、次のように近似することができる。

【0099】

$$C = \text{hopt} \cdot 2L = 1.69 \times (2 \times 23) = 77.7$$

ただし、

hopt=1.69ホップ

L=23リンク

N=11ノード

l=4.18(46/11)リンク/ノード

【0100】平均ホップ数は、ノード間のすべての最適（最小）経路から決定される。

・1ホップ：AW、AT、AX、AB、BW、BX、C  
W、CX、FX、FZ、DE、DZ、DY、EZ、E  
Y、WT、WX、WZ、XZ、XY、XT、TY、Z  
Y。

・2ホップ：AC、AZ、AY、AF、BC、BZ、B  
Y、BF、BT、CZ、CY、CF、CT、FD、F  
E、FY、FT、FW、DX、DT、DW、EX、E

50

T, EW, ZT, YW.

・3ホップ: AE, AD, BD, BE, CD, CE

$hopt = (1 \times 23 + 2 \times 26 + 3 \times 6) / (23 + 26 + 6) = 93 / 55 = 1.69$  ホップ/最適経路 \*

$C = hopt \cdot 2L' = 1.69 \times (2 \times 13) = 43.9$

ただし、

・ $L' = 13$  リンク

・ $hopt = 1.69$  ホップ

【0102】この複雑さの低減は、典型的なネットワークにおいて非常に重要となり得る。

【0103】まとめとして、本発明の構成に関して以下の事項を開示する。

(1) 伝送リンクと相互接続された複数のノード(201ないし208)とを含むバケット交換通信ネットワークにおいて、該ネットワークのアクセス・ノードが、データ・バケットを受信し送信する手段と、ネットワーク構成要素を記憶し更新する手段とを含むとともに、ネットワーク内に位置する各宛先ノードと通信するための1または複数の使用可能リンクを事前選択して該事前選択の情報を記憶するリンク事前選択手段と、各接続要求ごとに、前記記憶された事前選択リンクのうち、前記アクセス・ノードから宛先ノードに至る最適指定経路を決定する手段とを備える、通信ネットワーク。

(2) 前記記憶更新手段が、前記アクセス・ノードをバックボーン・ノードまたはローカル・ノードとして識別する手段と、前記ネットワークの他のノードについてバックボーン・ノードとローカル・ノードとを識別する手段と、バックボーン・リンクとローカル・リンクとを識別する手段と、を含む、上記(1)に記載のネットワーク。

(3) 前記リンク事前選択手段が、すべてのバックボーン・リンクを、経路決定に使用可能なものとして選択する手段と、ノードまたは宛先ノードあるいはその両方に結合されたすべてのリンクを、経路決定に使用可能なものとして選択する手段と、他のすべてのリンクを、経路決定に使用可能なものとして選択しない手段とを含む、上記(1)または(2)に記載のネットワーク。

(4) 前記指定経路決定手段が、修正ベルマン・フォード・アルゴリズムを計算する手段を含む、上記(1)、

(2)または(3)に記載のネットワーク。

(5) アクセス・ノード(300)において、伝送リンク(209)と相互接続された複数のノード(201ないし208)とを含むバケット交換通信ネットワーク(200)内での指定経路を選択するための方法であって、ネットワーク構成要素を記憶し更新する段階を含むとともに、ネットワーク内に位置する各宛先ノードと通信するための1または複数の使用可能リンクを事前選択して該事前選択の情報を記憶するリンク事前選択段階と、各接続要求ごとに、前記記憶された事前選択リンクのうち、前記アクセス・ノードから宛先ノードに至る最

\*【0101】本発明による使用可能リンクの事前選択により、最小ホップ制約が低下することはないと仮定すると、この例におけるアルゴリズムの複雑さをほぼ1/2に減少させることができる。

適バケット指定経路を決定する段階とを含む方法。

(6) 前記記憶更新段階が、前記アクセス・ノード自身をバックボーン・ノードまたはローカル・ノードとして識別する段階と、前記ネットワークの他のノードについてバックボーン・ノード(402)とローカル・ノード(404)とを識別する段階と、バックボーン・リンク(403)とローカル・リンク(405)とを識別する段階とを含む、上記(5)に記載の方法。

(7) 前記リンク事前選択段階が、すべてのバックボーン・リンクを、経路決定に使用可能なものとして選択する段階と、ノードまたは宛先ノードあるいはその両方に接続されたすべてのリンクを、経路決定に使用可能なものとして選択する段階と、他のすべてのリンクを、経路決定に使用可能なものとして選択しない段階とを含む、上記(5)または(6)に記載の方法。

(8) 前記指定経路決定段階が、修正ベルマン・フォード・アルゴリズムを計算する段階を含む、上記(5)、(6)または(7)に記載の方法。

【0104】

【発明の効果】本発明にしたがって、ネットワークをバックボーン・ノードとローカル・ノードに分離することにより、ネットワークにおける経路選択が高速化される。

【図面の簡単な説明】

【図1】従来の技術においてL、クラインロックとF、カムーンが提唱した方法による、階層的経路指定を示す図である。

【図2】高速ネットワークによって支援される、様々な回線スループット能力の機能において必要な処理時間(1秒あたりの命令数)を示すグラフである。

【図3】本発明において特許請求されるアクセス・ノードおよび通過ノードを含む高速バケット交換ネットワークの典型的なモデルを示す図である。

【図4】本発明による高速経路指定点を示す図である。

【図5】本発明による、複数のバックボーン・ノードとローカル・ノードへのネットワークの分解を示す図である。

【図6】本発明による、ネットワーク内への新しいノードおよびリンクの追加と、トポロジ・データベースにおけるそれらの分類を示す図である。

【図7】ノードAの最適経路ツリーを示す図である。

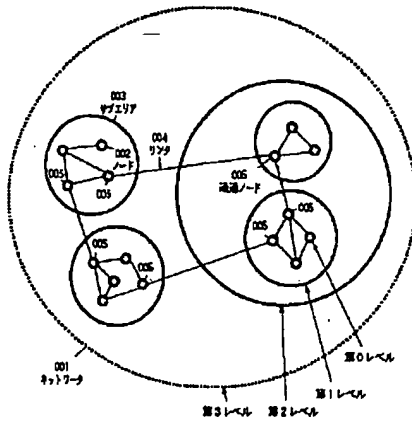
【図8】本発明によるノードAの最適経路ツリーを示す図である。

【図9】本発明による経路選択方法を要約したブロック図である。

## 【符号の説明】

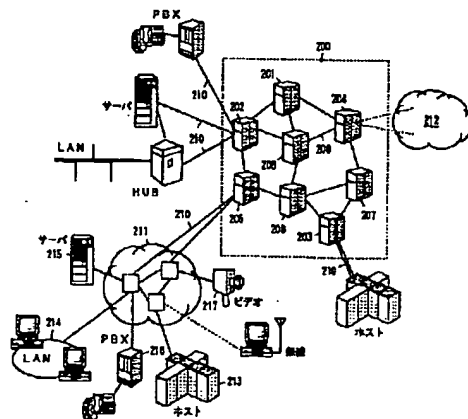
- 200 高性能ネットワーク
- 201 ノード
- 202 アクセス・ノード
- 203 アクセス・ノード
- 204 アクセス・ノード
- 205 アクセス・ノード
- 206 ノード
- 207 ノード
- 208 ノード

【図1】



階層的経路指定

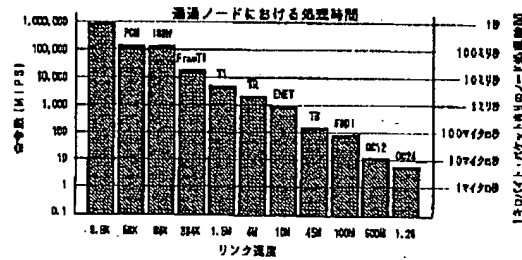
【図3】



高速パケット交換ネットワーク

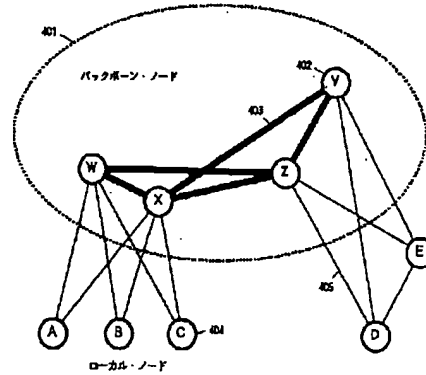
- 209 トランク
- 210 アクセス
- 211 リンク
- 212 ユーザ・ネットワーク
- 213 企業サーバ
- 214 ローカル・エリア・ネットワーク (LAN)
- 215 アプリケーション・サーバ
- 216 構内交換機、(PBX)
- 217 ビデオ・サーバ

【図2】



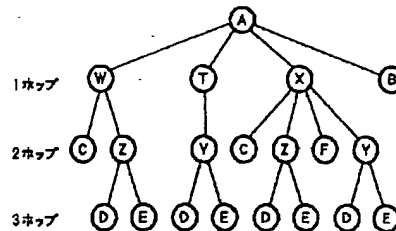
【図5】

ネットワークの分解



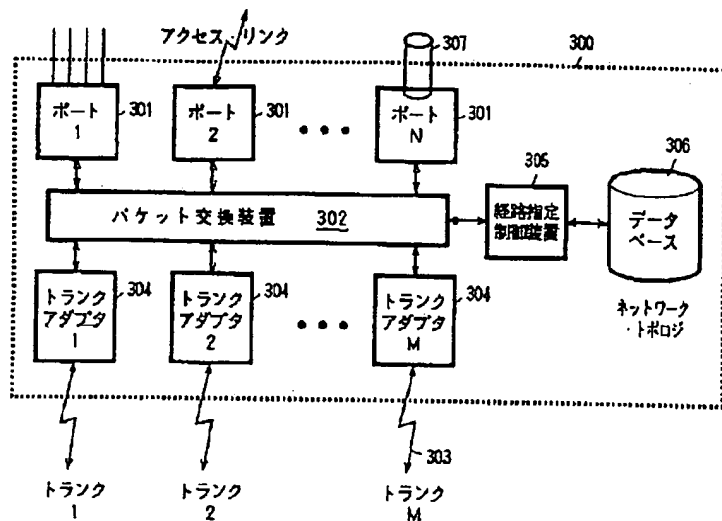
【図7】

ノードAの最短経路ツリー



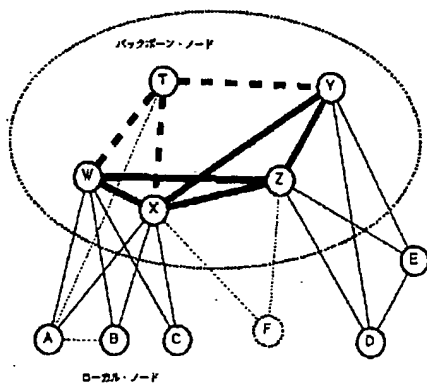
【図4】

## 高速バケット交換ノード



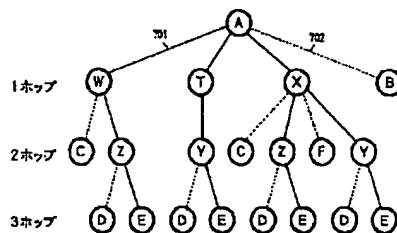
【図6】

## ネットワークの更新

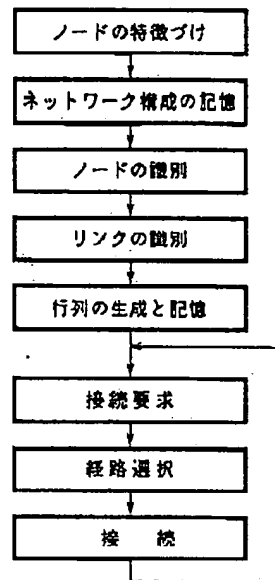


【図8】

## ノードAからノードEにいたる最速経路



【図9】



伝送リンクを事前選択する方法

フロントページの続き

(72)発明者 フロード・ガラン  
 フランス06800 カーニュ・シュル・メー  
 ル アヴェニュー・ド・テュイリエール56

(72)発明者 ジェラルール・ルビビー  
 フランス06640 ヴァンス アヴェニュー・  
 ド・ボワリュ113  
 (72)発明者 オリヴィエ・モーレル  
 フランス06110 ル・キャヌ リュ・フォ  
 ルヴィーユ15